AMENDMENTS TO THE CLAIMS

(IN FORMAT COMPLIANT WITH THE REVISED 37 CFR 1.121)

1. (CURRENTLY AMENDED) An apparatus comprising:

a processor (i) comprising a number of internal registers and (ii) configured to manipulate contents of said internal registers in response to instruction codes of a first instruction set;

a processor interface circuit coupled to said processor;

a memory interface circuit coupled to a memory device;

an extension stack coupled between said processor interface and said memory interface; and

a translator circuit (i) coupled between said processor interface and said memory interface and (ii) configured to implement a stack using one or more of the internal registers of said processor and said extension stack.

2. (PREVIOUSLY PRESENTED) The apparatus according to claim 1, wherein said one or more internal registers are used to store a top of said stack.

3. (PREVIOUSLY PRESENTED) The apparatus according to claim 2, wherein said top of said stack comprises a Java virtual machine (JVM) top of stack (TOS).

4. (ORIGINAL) The apparatus according to claim 1, wherein said internal registers are dynamically allocated in response to stack status.

5. (PREVIOUSLY PRESENTED) The apparatus according to claim 1, wherein said translator circuit is further configured to generate one or more instruction codes of the first instruction set for controlling the internal registers in response to an instruction code of a second instruction set.

6. (PREVIOUSLY PRESENTED) The apparatus according to claim 5, wherein said instruction code of said second instruction set comprises a stack instruction.

7. (CURRENTLY AMENDED) The apparatus according to claim 1, wherein said translator circuit comprises a stack management unit coupled to said processor interface, said memory interface and said extension stack.

8. (CURRENTLY AMENDED) The apparatus according to claim 7, wherein said stack management unit is configured to control transfers between (i) said extension stack and ~~a~~ said memory device and (ii) said internal registers and said extension stack.

9. (PREVIOUSLY PRESENTED) The apparatus according to claim 1, wherein said extension stack is implemented as a last-in first-out (LIFO) memory.

10. (PREVIOUSLY PRESENTED) The apparatus according to claim 1, wherein said extension stack comprises both head and tail interfaces.

11. (PREVIOUSLY PRESENTED) The apparatus according to claim 8, wherein said extension stack (i) is emptied to said memory device to prevent an overflow and (ii) filled from said memory device to prevent an underflow.

12. (ORIGINAL) The apparatus according to claim 11, wherein said memory device comprises a main memory of said processor.

13. (ORIGINAL) The apparatus according to claim 7, wherein said extension stack is configured to indicate an almost empty or almost full condition.

14. (PREVIOUSLY PRESENTED) The apparatus according to claim 7, wherein said stack management unit is further configured to track which internal registers are used for the stack.

15. (ORIGINAL) The apparatus according to claim 14, wherein said stack management unit is further configured to track how many internal registers are used for the stack.

16. (PREVIOUSLY PRESENTED) The apparatus according to claim 7, wherein said stack management unit is configured to control (i) pushes to said one or more internal registers from said extension stack and (ii) pops from said one or more internal registers to said extension stack.

17. (CURRENTLY AMENDED) An apparatus comprising:

means for manipulating data in response to instruction codes of a first instruction set, said manipulating means comprising a number of internal registers; and

means for translating instruction codes of a second instruction set into sequences of said instruction codes of said first instruction set, wherein said translating means is configured to implement a stack with (i) one or more of said internal registers and an extension stack coupled between said manipulating means and a memory device, (ii) use said one or more of said internal registers as a top of stack, (iii) empty said extension stack to ~~a~~ said memory device, (iv) refill said extension stack from said memory device, (v) transfer contents of said one or more internal registers to said extension stack and (vi) transfer

contents of said extension stack to said one or more internal registers.

18. (CURRENTLY AMENDED) A method for implementing a Java virtual machine top of stack comprising the steps of:

(A) translating one or more instruction codes of a first instruction set into sequences of instruction codes of a second instruction set;

(B) manipulating contents of one or more internal registers of a processor in response to said sequence of instruction codes of said second instruction set; and

(C) implementing a stack comprising said one or more internal registers and an extension stack coupled between said processor and a memory device, wherein said one or more internal registers are configured as a top of stack.

19. (ORIGINAL) The method according to claim 18, wherein said instruction codes of said first instruction set comprise stack operations.

20. (CURRENTLY AMENDED) The method according to claim 18, further comprising the step of:

transferring values between said internal registers and said extension stack in response to a first one or more of said sequences of instruction codes of said second instruction set; and

transferring values between said extension stack and ~~a~~ said memory device in response to watermark indications from said extension stack.

21. (CURRENTLY AMENDED) The method according to claim 18, further comprising the step of:

generating control signals configured to (i) empty said extension stack to ~~a~~ said memory device in response to a high watermark of said extension stack being reached, (ii) refill said extension stack from said memory device in response to a low watermark of said extension stack being reached, (iii) transfer values from said one or more internal registers to said extension stack and (iv) restore values from said extension stack to said one or more internal registers.

22. (CURRENTLY AMENDED) The apparatus according to claim 1, wherein:

said translator circuit is configured to (i) empty said extension stack to ~~a~~ said memory device in response to an overflow condition of said extension stack, (ii) refill said extension stack from said memory device in response to an underflow condition of

said extension stack, (iii) transfer contents of said one or more internal registers to said extension stack in response to an overflow condition of said one or more internal registers and (iv) transfer contents of said extension stack to said one or more internal registers in response to an underflow condition of said one or more internal registers.

23. (CURRENTLY AMENDED) The apparatus according to claim 1, further comprising:

a register block coupled between said processor <u>interface</u> and said extension stack and configured to operate as a bridge between said processor and said extension stack.